The Solution of Simultaneous Ordinary Differential Equations Using a General Purpose Digital Computer

W. H. Anderson, *Bendix Products Division-Missiles, Bendix Aviation Corporation, Mishawaka, Indiana*

Introduction

Scientific problems are quite often expressed mathematically in the form of ordinary differential equations. In many cases, these equations are nonlinear and cannot be solved analytically. Fortunately, their solution can often be obtained by means of analog or digital computing equipment. Analog simulations will not be discussed here except to state that they are very good in some applications but poor in others where the size of the problem and accuracy requirements prohibit their use.

In order that differential equations may be solved on a general purpose digital computer, a method of numerical integration must be selected and included in the program. The problem of choosing the best method will be avoided here except to the extent that a means of comparing different methods will be indicated. The method of Runge Kutta including error checking and interval modification will be discussed in detail.

The Problem

The analysis of scientific problems frequently leads to a set of simultaneous ordinary differential equations which describe the phenomenon being studied. These equations are usually nonlinear and cannot be solved analytically. Approximations can sometimes be made in an attempt to linearize the equations, but this is often undesirable, as significant errors may be introduced.

While equations for different problems vary in number, order, and size, they can be manipulated mathematically for any given problem into the general form:

$$\begin{aligned}
\dot{y}_1 &= f_1(t, y_1, y_2, \cdots, y_m) \\
\dot{y}_2 &= f_2(t, y_1, y_2, \cdots, y_m) \qquad (1) \\
&\vdots \\
\dot{y}_m &= f_m(t, y_1, y_2, \cdots, y_m)
\end{aligned}$$

where

$$\dot{y}_1 = \frac{dy_1}{dt}, \quad \text{etc.}$$

The solution of the set of equations (1) is a tabulation of values of the dependent variables y for various values of the independent variable t when given the initial values:

$$\begin{aligned}
t &= t_n \\
y_1 &= y_1(t_n) \\
y_2 &= y_2(t_n) \\
&\vdots \\
y_m &= y_m(t_n)
\end{aligned}$$

The Solution

The determination of the desired values of y involves the integration of the differential equations (1). The work involved in performing this integration can be greatly reduced by the use of analog or digital computers.

Analog computers serve the purpose very well in quite a few applications. However, there are problems which cannot be solved satisfactorily on the analog computer because the problem may be too large or extreme accuracy may be desired. In cases such as these, a digital simulation is required.

The programming of a problem of this type for a digital computer can be greatly simplified by breaking the problem up into two sections, one of which is the integration routine and the other being the function evaluation. These two programs may be written independently of each other once a few basic rules are set forth specifying the functions that each program is to perform.

Machine storage must be allocated for each of the two programs and the condition of the various accumulators must be specified upon entry to the exit from either section. For the purpose of this discussion, the following definitions and rules will be used:

REGION—a section of machine storage containing a specified number of consecutive locations addressable symbolically by an alphabetic letter, followed by an R, followed by a number which specifies the particular word in the region. Example: BR12 is word 12 in region B.

REGION I —reserved for the integration routine program
REGION F—reserved for the function evaluation program
REGION P—reserved for precision
REGION Y—reserved for the dependent variables
REGION D—reserved for the derivatives of the dependent variables
t —independent variable
h —increment of the independent variable or integration interval
IR1 —entry to initialize integration routine
FR1 —entry to initialize function evaluation
IR2 —entry to integration routine from FR1 loop
FR2 —entry to function evaluation for purpose of evaluating equations (1)
IR3 —entry to integration routine from FR2 loop
FR3 —entry to function evaluation for output purposes
IR4 —entry to integration routine from FR3 loop
m —number of differential equations
PRECISION—1/*allowable error per unit of independent variable*

The integration routine is responsible for the following:

(a) Reading in initial values (t, h, y_1, y_2, $\cdots$, y_m) from cards or tapes, etc., and placing the dependent variables in REGION y. This is done in the initializing loop which starts at location IR1.

(b) Performing a step-by-step integration using some numerical method. This is done by obtaining values of the derivatives of y_1, y_2, $\cdots$, y_m at t by entering the function evaluation at location FR2. When the integration routine is re-entered at location IR3, the values of $\dot{y}_1$, $\dot{y}_2$, $\cdots$, $\dot{y}_m$ are in locations DR1, DR2, $\cdots$, DRm, respectively.

(c) Performing truncation error checking on every y variable each time a new point is found. (I.e.: Assume that the integration has proceeded to the point t_n and that the y variables and their derivatives are known at t_n, $t_n - a_1h$, $t_n - a_2h$, etc. An integration formula is now used to find the values of the y variables at $t_n + a_0h$. The truncation error which is introduced by the integration method over the inverval a_0h must be computed for each of the y variables and checked against their respective precisions which are located in PR1, PR2, $\cdots$, PRm. If the truncation error exceeds the allowable error specified by the precision for any of the y variables, the values at $t_n + a_0h$ cannot be accepted. The interval of integration must be reduced to a value which gives truncation errors which are less than the allowable errors specified for each of the y variables.)

(d) When a new point is found and is acceptable as described in (c), a new interval of integration should be chosen for the next step. The difference between the truncation error and the allowable error over the last interval provides sufficient information for the determination of the proper interval size for optimum operation. This difference is computed for all of the y variables, and the smallest one is used to determine the new interval. Modification of the interval may not be possible at each step when using certain methods of numerical integration, such as Milne. However, sufficient past history of the variables should be stored so that the interval may be doubled if the error check indicates that the solution will remain within the accuracy limitations.

(e) After a new point has been accepted and the next interval chosen, the function evaluation should be entered at location FR3 to execute the output program. At this point, the solution has proceeded to a time t_n, the y variables are in REGION Y, their derivatives are in REGION D and the new interval is in h. When the integration routine is re-entered at location IR4, the necessary initialization is done so that the solution may proceed to the next point.

The function evaluation is responsible for the following:

(f) Initialization of the function evaluation program. This loop begins at location FR1 and exits to location IR2 upon completion. The initial values of the y variables are in REGION Y, time is in t, and the integration interval is in h when this program is executed.

(g) Evaluation of the functions given by equations (1) using the y variables in REGION Y and the time in t. The computed values f_1, f_2, $\cdots$, f_m are to be placed in DR1, DR2, $\cdots$, DRm, respectively. This program begins at location FR2 and exits to location IR3 upon completion.

(h) Preparation of an output program which punches, prints, etc., the desired variables in the proper format. The starting location of this program is FR3, and exit is to location IR4 upon completion. This program should also determine when the solution has been completed. It may be desirable to modify the interval of integration in this program in order that the solution may be computed at a particular value of the independent variable t. It is possible to do this when using methods of integration such as Runge Kutta, but special techniques must be applied for predictor corrector type methods.

Discussion

The problem of determining the solution of the differential equations (1) has been broken up into two sections, and the area of responsibility of each section has been defined. However, there are still a few details which must be specified, but these depend on the particular machine for which the program is written.

The reading in of the initial values is to be programmed in the initializing loop of the integration routine. An input format must therefore be specified so that this program may obtain the data and place it in the proper areas. Once this format has been set up, the programmer of the function evaluation must prepare his initial values accordingly.

There are many methods of numerical integration which lend themselves nicely to this type of programming, and it is recommended that several different methods be programmed compatibly so that they may be used interchangeably with any given function evaluation. It is then possible to check the various methods on particular problems and select the program which is most efficient in each case.

The truncation error checking is one of the most important features of this system. Not only does it prevent an excessive amount of error in the solution, but it is also helpful in checking out the function evaluation program.

A common mistake in function evaluation programs is the entering of erroneous parameters or points in a table. This type of error generally shows up as a step function, which may be extremely difficult to integrate if the error is significant. If this is the case, the integration routine will decrease the interval to such a point that it will be obvious to the programmer that an error has been made. If automatic error checking is not included in the integration routine, the tendency is to use a much smaller interval than is necessary to gain the desired accuracy. This may easily lead to a factor-of-ten increase in computing time, and the solution is less likely to be correct.

The difference between the allowable error and the truncation error is an indication of the amount of machine time being wasted. When this difference is computed for each of the y variables in the error checking loop, the smallest difference should be retained for interval modification purposes. Assuming that the truncation error will vary over the next interval only as a function of h, it is not difficult to determine a value of h which gives an error difference which is closer to zero than the current one. Selecting an interval which would reduce the difference to zero is not feasible since, if the errors increase slightly, the next point will not be accepted.

It is suggested in (h) that the output program may want to select the integration interval at times. If Runge Kutta integration is used, the output program should be able to merely replace h with the desired value. If this procedure is to be allowed in the output program, methods which rely on several previous points at an even interval will be somewhat inconvenienced. When using a method of this type, the interval should be checked in the IR4 loop. If it has been changed in the output program, special care should be taken to insure that the solution includes the point t_n+h, h being the interval selected in the output program. Since special starting procedures are required for methods of this type, the solution may be considered as a new problem and a restart made in the integration routine using an interval that allows the values at t_n+h to be computed. The output program should not be entered again after it has modified the interval until the variables have been computed at the time it has selected.

The system which has been described here for solving differential equations is by no means unique. The responsibilities of each of the two programs may be varied depending on the computer to be used, the type of problems to be solved, and the capability of the user from an engineering as well as a programming standpoint. It may be desirable to set up the integration program in more of a subroutine fashion than has been outlined here.

The problem of selecting the best numerical method for use in the integration routine is indeed a difficult one. There are many different methods which may be used, and each method usually has particular applications to which it is well suited. One approach is to write several integration programs using a different method of integration in each program. When a function evaluation program is written for a given set of differential equations, the integration routines may be interchanged to determine the most efficient combination.

When using predictor-corrector-type methods of integration, the truncation error may be expressed as a function of the difference between the predicted and corrected values and is thus readily computed. The calculation of the truncation error for Runge Kutta integration is not as straightforward. A means of computing truncation error as well as a flow chart for a Runge Kutta integration routine are given in the Appendix.

Appendix

The following method for computing Runge Kutta truncation error appears to be extremely inefficient as it requires a minimum of eleven function evaluations over two integration intervals. However, an integration routine which incorporates this method was found to be more efficient, for a majority of the problems encountered, than a program which uses Milne integration. This was due to the average interval ratio being larger than eleven to one.

Runge Kutta Truncation Error

For the purpose of this discussion, consider the differential equation

$$\dot{y} = f(y,t)$$

where $y = y_0$ when $t = t_0$. Using the Runge Kutta method with an interval of h, the value

$$y = y_1$$

when

$$t = t_0 + h = t_1$$

can be found. With the initial values (y_1,t_1) the integration now proceeds to the point:

$$y = y_{2,1}$$

when

$$t = t_1 + h = t_2 .$$

The value of y at t_2 is now computed using the initial values (y_0,t_0) with an interval of $2h$:

$$y = y_{2,2}$$

when

$$t = t_0 + 2h = t_2$$

Since Runga Kutta is a fourth order method, there is a fifth order error introduced over each integration step. Assuming that this error is strictly proportional to h^5 from t_0 to t_2, the following errors may be defined:

Error in y_1 : $E_1 = h^5k$

Error in $y_{2,1}$: $E_{2,1}$ = error introduced from t_0 to t_1 plus error introduced from t_1 to t_2

$$E_{2,1} = 2h^5k$$

Error in $y_{2,2}$: $E_{2,2} = (2h)^5k$

If the desired value at t_2 is designated by y_2, the values which have been computed may be written as:

$$y_{2,1} = y_2 + E_{2,1} = y_2 + 2h^5k$$

$$y_{2,2} = y_2 + E_{2,2} = y_2 + 32h^5k$$

To find $E_{2,1}$: $y_{2,2} - y_{2,1} = 30h^5k = 15E_{2,1}$

or : $E_{2,1} = \dfrac{y_{2,2} - y_{2,1}}{15}$

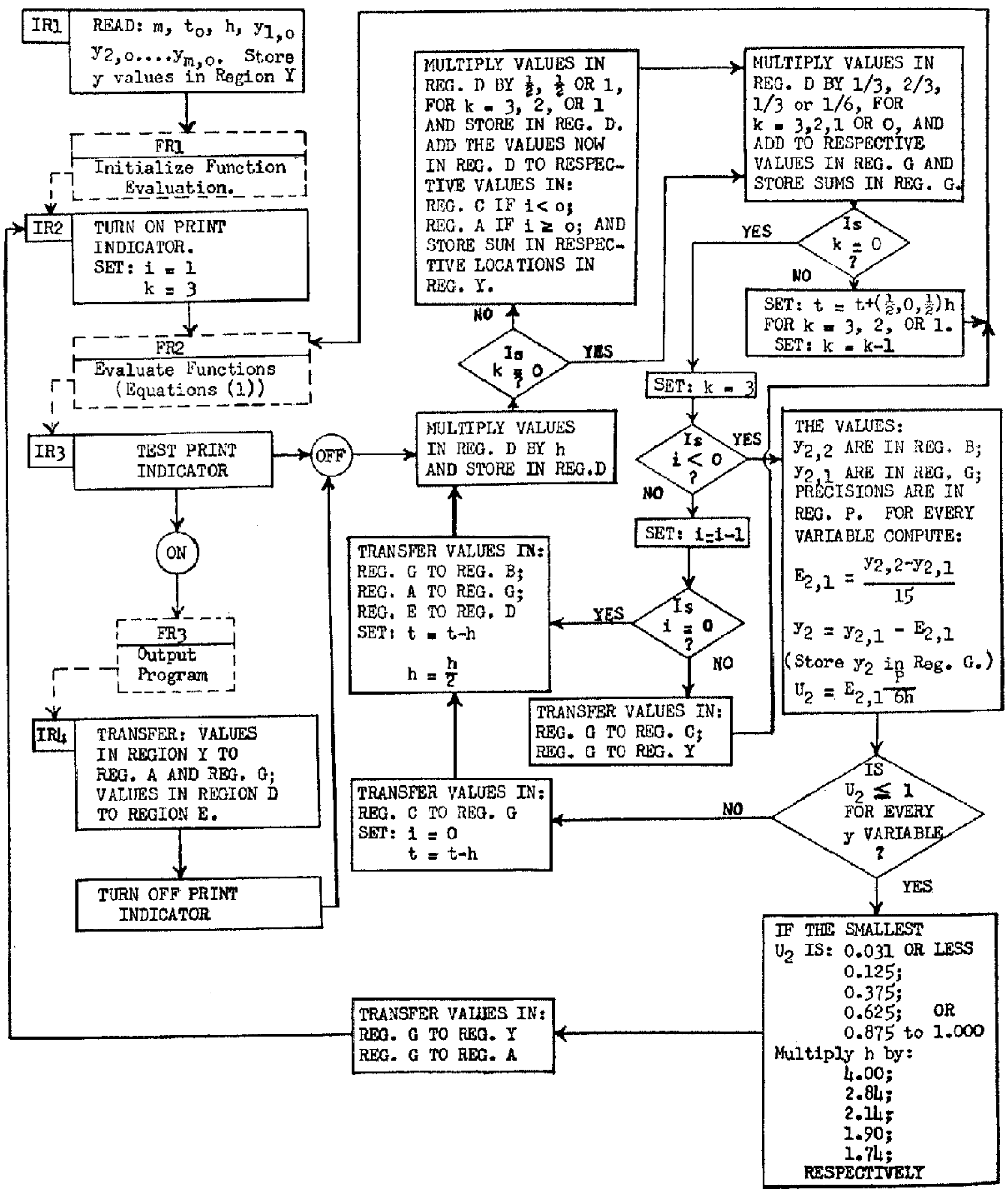

FIG. 1

The error in $y_{2.1}$ can thus be calculated and subtracted from $y_{2.1}$ to get a better value of y_2:

$$y_2 = y_{2.1} - \frac{y_{2.2} - y_{2.1}}{15}$$

There will still be, in general, some truncation error in y_2. While it will be less than the calculated value of $E_{2.1}$, the exact amount will vary depending on the function being integrated. A factor of $\frac{1}{3}$ has been used and found to be satisfactory. The truncation error in y at t_2 is thus:

$$E_2 = \frac{1}{3} \cdot \frac{y_{2.2} - y_{2.1}}{15}$$

If the allowable error per unit of independent variable is specified by A, and the precision P is defined by

$$P = \frac{1}{A},$$

the allowable error over the double interval is

$$2hA = \frac{2h}{P}.$$

This error may be divided into E_2 to obtain the ratio

$$U_2 = \frac{\frac{1}{3} \cdot \frac{y_{2.2} - y_{2.1}}{15}}{\frac{2h}{P}} = \frac{1}{45} \frac{P}{2h} (y_{2.2} - y_{2.1})$$

Under the above assumptions, if U_2 is less than one, the error made in y_2 is less than the allowable error.

Substituting the relation

$$y_{2.2} - y_{2.1} = 30h^5k$$

into the above expression, U becomes

$$U_2 = \frac{P}{3} h^4k.$$

Using (t_2, y_2) as initial values, the integration may now proceed to the point (t_4, y_4). However, if U_2 is much less than one, it is desirable to use a larger interval from t_2 to t_4 than was used from t_0 to t_2. When y_4 has been computed, U at that point will be

$$U_4 = \frac{P}{3} h_p^4 k_4$$

where

$$h_p = \frac{1}{2}(t_4 - t_2).$$

Thus

$$\frac{U_4}{U_2} = \frac{\frac{P}{3} \cdot h^4 p k_4}{\frac{P}{3} \cdot h^4 k}.$$

It was assumed above that the truncation error is proportional to h^5 from t_0 to t_2. If it is now assumed that the truncation error is proportional to h^5 from t_0 to t_4, then

$$k = k_4$$

and

$$\frac{U_4}{U_2} = \frac{h^4 p}{h^4}$$

or

$$h_p = h \left(\frac{U_4}{U_2}\right)^{1/4}.$$

Since the values of h and U_2 are known at t_2, the new interval h_p may be determined by specifying a desired value for U_4. The optimum value is one, but since a rerun of an interval would be required if U_4 obtained exceeds one, a more conservative specification is indicated. A value of $U_4 = 0.5$ has been used and found to be satisfactory. The integration interval from t_2 to t_4 is calculated by

$$h_p = \left(\frac{0.5}{U_2}\right)^{1/4} h.$$

To eliminate the use of a subroutine in evaluating $(0.5/U_2)^{1/4}$, a table of this function for various values of U_2 may be stored in the integration routine. This results in a saving of time and storage and is sufficiently accurate considering the assumptions that were made.

The following table has been used and found to be satisfactory:

U_2	$(0.5/U_2)^{1/4}$
0.031 or less	2.00
0.125	1.42
0.375	1.07
0.625	0.95
0.875 to one	0.87

It is felt that any factor greater than two is not required, as the use of this method should keep the value of U in the mid-range. However, this is a function of the type of problems being solved, and a different table may prove more economical under different circumstances.

In the event that U_2 is found to be greater than one when the error check is performed, the interval is cut in half and the solution computed at t_1. The error check must then be made at this point to insure the desired accuracy.

The flowchart (Fig. 1) demonstrates the manner in which a Runge Kutta integration routine might be programmed for a set of m simultaneous differential equations.

The regions:

A—used for storing initial values y_0 for all variables
B—used for storing the values $y_{2.2}$ for all variables
C—used for storing the values y_1 for all variables
E—used for storing the values $\dot{y}_0$ for all variables
G—used for storing partially computed y_{n+1} (defined below) for all variables

all require m locations and are within REGION I. The other regions and notations are consistent with the text. One point, which may be considered a deviation from the derivation, is that the first integration using an interval h obtains the point $y_{2,2}$. Then the interval is cut in half to obtain the points y_1 and $y_{2,1}$ successively. This is necessary if the point y_1 is to be retained for use as $y_{2,2}$ in the event that the errors are excessive, requiring that the interval be again cut in half.

It is noted that REGION E may be eliminated if storage is a problem. If REGION E is not used, and extra evaluation is required making a total of 12 evaluations per step.

The equations of Runge Kutta are

$$y_{n+1} = y_n + \tfrac{1}{6}(C_1 + 2C_2 + 2C_3 + C_4)$$

where

$$C_1 = hf(t_n, y_n)$$

$$C_2 = hf\left(t_n + \frac{h}{2}, y_n + \frac{C_1}{2}\right)$$

$$C_3 = hf\left(t_n + \frac{h}{2}, y_n + \frac{C_2}{2}\right)$$

$$C_4 = hf(t_n + h, y_n + C_3).$$

THE FUTURE OF AUTOMATIC DIGITAL COMPUTERS (cont'd from page 341)

process is that the inherently great magnification of the electron microscope makes the reading of closely packed data easy; another advantage is that the reading element is an electron beam which can be deflected electrically and thus facilitates high speeds of operation. Naturally the advantages of non-destructive read-out are still present, as with the magnetic drum.

Even if this were the only feature of Meyer's work it would be exciting, but he has also shown [7] how information may be written on to a magnetic medium by means of an electron beam. This is achieved by magnetising the medium initially by means of an external field, and then heating small regions above the Curie point by means of a sharply focused electron beam. When the material cools below the Curie point again, an inversion effect takes place and this enables the electron mirror microscope to detect the previously heated spots. Meyer suggests that information densities of 10^6 bits/cm^2 will be possible and that the writing speed might be 10^5–10^6 bits/sec.

In the general field of input-output equipment clearly the magnetic tape will come into general use. Xerography, too, is likely to be applied both to printing and to replacement of punched paper or film. A second form of recording, "Digitape" based on "Teledeltos" paper, is, I think, less likely to be perpetuated because of its expensive basic material. Devices for the direct reading of printed characters are at present moderately unreliable. The imperfections of print, and especially of typewritten print, present quite considerable technical difficulties to the designer of a machine which, independently of context, will recognise any given printed character with great accuracy. By using context the problem is rendered easier and such an extension of the basic character reader is almost inevitable. Direct recognition of the spoken word is on the horizon. One possible application is to stocktaking, where a pocket recorder and lapel microphone would leave hands free for inspecting the stock.

My feeling on all questions of input-output is, however, the less the better. The ideal use of a machine is not to produce masses of paper with which to encourage Parkinsonian administrators and to stifle human inventiveness, but to make all decisions on the basis of its own internal operations. Thus computers of the future will communicate directly with each other and human beings will only be called on to make those judgements in which aesthetic considerations are involved.

REFERENCES

1. Burks, A., Goldstine, H. H., and von Neumann, J. *Preliminary Discussion of the Logical Design of an Electronic Instrument.* Princeton, 1946.
2. Booth, A. D., and Britten, K. H. V. *General Considerations in the Design of an All Purpose Electronic Digital Computer.* Princeton, 1947.
3. Booth, A. D., and Booth, K. H. V. *Automatic Digital Calculators*, 2nd ed., p. 145, Butterworths, 1956.
4. Bradley, E. M. Private communication.
5. Buck, D. A. *Proc. I.R.E.*, Vol. 44 (1956), pp. 482–93.
6. Meyer, L. Electron Mirror Microscopy of Patterns Recorded in Magnetic Tape, *J. Appl. Phys.*, Vol. 29 (1958), pp. 658–60.
7. Meyer, L. Magnetic Writing with an Electron Beam, *J. Appl. Phys.*, Vol. 29 (1958), pp. 1454–6.